Improving the Observing Efficiency of Hubble Space Telescope

Mark D. Johnston, Ron Henry, Andrew Gerb, Mark Giuliano, Brian Ross, Nick Sanidas, and Steve Wissler
Space Telescope Science Institute, Baltimore, Maryland USA
and
Jim Mainard
TRW Inc., Los Angeles, California USA

Abstract

This paper summarizes the status of the technical effort to improve the scheduling (observing) efficiency of Hubble Space Telescope. It focuses on the software systems and tools which are used by the operations staff to prepare, plan, and schedule HST observations. A set of high-leverage improvements has been identified and implemented. A comprehensive test program was defined and executed to measure the potential scheduling efficiency of the combined HST ground system elements. This test program has demonstrated that the software improvements have achieved their goal of enabling high-efficiency operation.

1.0 Introduction

Hubble Space Telescope (HST) was launched into a low-earth orbit in April 1990 as the first of NASA's Great Observatories. Science operations for the observatory are conducted at Space Telescope Science Institute in Baltimore. Although the spherical aberration of the HST's primary mirror, discovered just after launch in mid-1990, has seriously limited the capabilities of the telescope, nevertheless HST has already demonstrated that it is a first-class research instrument and has been the source of a steady stream of highly significant astronomical results. The HST Servicing Mission, planned for December 1993, will install new instruments and corrective optics that will restore the performance of the observatory to very close to its originally planned level.

Competition for observing time on HST is severe. The available time is oversubscribed by a large factor during each of the annual observing program selections. As a result, it is extremely important that time on the Hubble be utilized as effectively as possible. Since the operation of the spacecraft is nearly entirely pre-planned (only small real-time pointing adjustments are permitted), the responsibility for efficient utilization of the telescope falls on the ground software systems and the people who operate them.

As may be expected, priorities during the early post-launch stages of the HST mission focused on establishing the safe and smooth operation of the observatory and on completing the software support for its most significant scientific capabilities. Once this was successfully accomplished, attention was shifted to observing efficiency. Following an analysis of ground system operations, a broad set of improvements was defined, implemented, tested, and installed in the operational system.

The following sections describe first the approach taken in each software area (Section 2.0), then the test program and results (Section 3.0), and finally the conclusions drawn from the successful completion of this project (Section 4.0). Appendix A contains a glossary of the terms and acronyms relevant to the HST science operations ground systems.

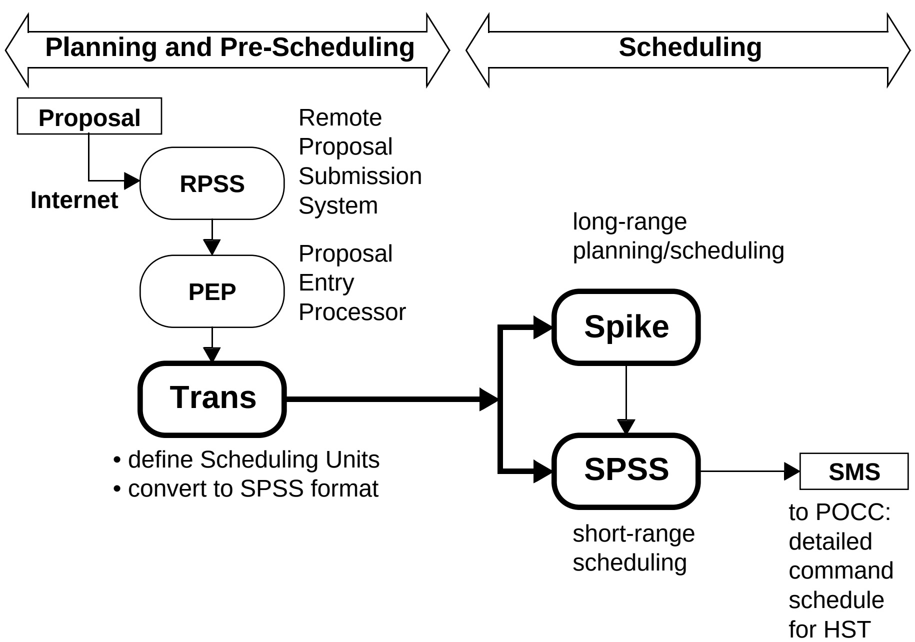
2.0 Approach

The goal of the efficiency improvements described here is to accomplish the most science in any given amount of elapsed time without compromising quality. It is clear that to accomplish this requires that "gaps" (wasted time) in the schedule must be minimized or eliminated. The basic scheduling entity in the HST ground system is called a *Scheduling Unit* or SU: a sequence of exposures and other spacecraft activities which is scheduled as a single entity. The key elements of the approach are:

1. **Define efficient-to-schedule SUs** by using all available knowledge and flexibility to eliminate dead-time gaps and unnecessary target occultations.
2. **Identify the times when these SUs can be efficiently scheduled** during the long-range planning interval, then provide a way to exploit this information when building the long-range plan.
3. **Improve the scheduling algorithms used for sequencing SUs** in the short-term schedule, so that higher-efficiency sequences can be constructed automatically.

These three elements most directly affect the *Transformation*, *Spike*, and *SPSS* software systems, respectively. These systems are described in more detail in references 1–5. Figure 1 illustrates the HST scheduling process flow and shows the central location of these three systems. Transformation is run first to convert HST observing proposals into the data structures required by both long-range planning and short-term scheduling. The long-range planning system is Spike, which commits SUs to weeks over a

FIGURE 1. Overview of the HST planning and scheduling systems. The efficiency improvements described here focused on the Transformation (Trans), Spike, and SPSS software systems, highlighted in the diagram below. These systems are most directly responsible for HST's achievable observing efficiency.



planning interval of a year or more. Short-term scheduling is done by the Science Planning and Scheduling System (SPSS) on a week-by-week basis, leading ultimately to the Science Mission Specification (SMS), the detailed command request schedule.

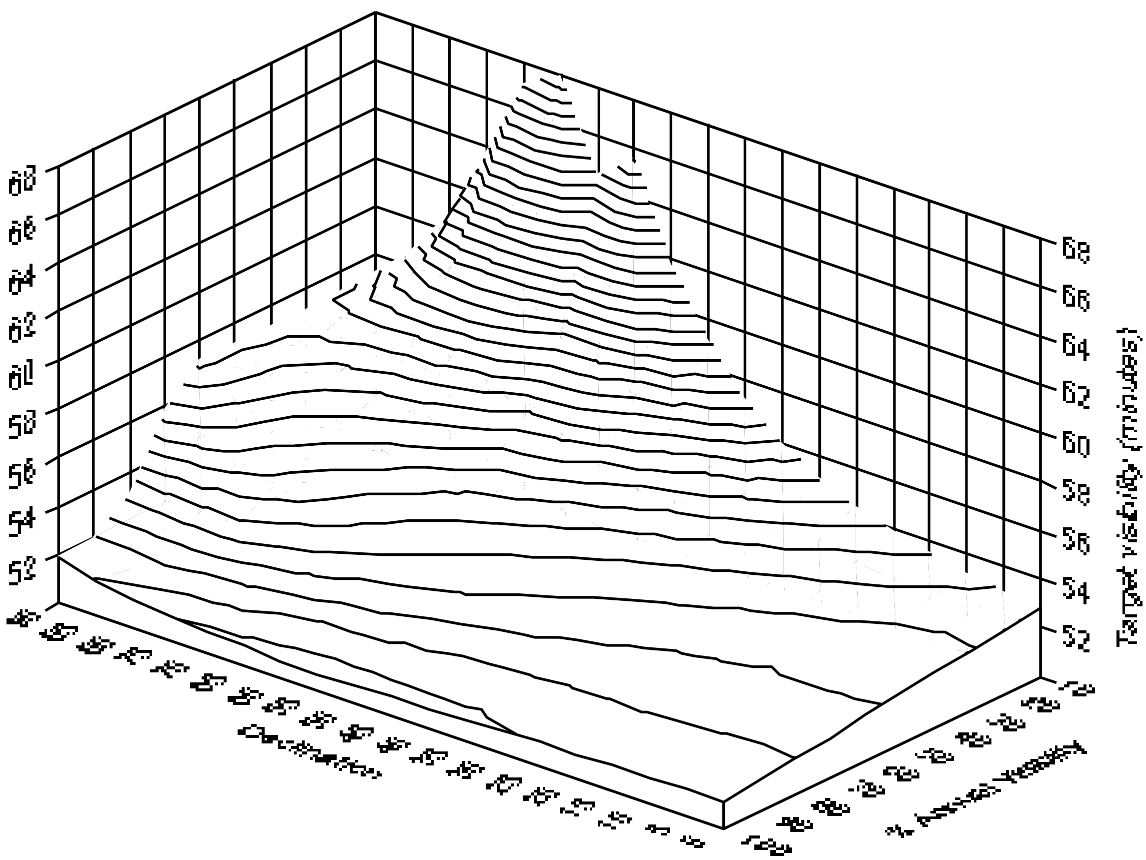
These three systems are strongly coupled: Transformation must define SUs which Spike can confidently allocate to efficient weeks and which SPSS finds possible to efficiently schedule. It is easy to imagine a more efficient SU which is also much harder to schedule (e.g. if the assumed target visibility intervals are slightly too short). It is clear that changes to any of the three systems cannot be made without carefully considering the impacts on the other systems and on their users. This has been an overriding concern during the implementation of the efficiency improvements, as will become clear below.

The following sections describe briefly the nature and rationale of the changes made to each of the Transformation, Spike, and SPSS software systems.

2.1 Transformation:Building a better Scheduling Unit

Except for one of the instruments known as the GHRS (Goddard High-Resolution Spectrograph), Scheduling Units consist entirely of fixed-size components which must ultimately be fit by SPSS into the available time around occultations and South Atlantic Anomaly (SAA) crossings. The GHRS can be treated as “interruptible” and can be scheduled by SPSS so as to essentially “flow around” unusable time periods. While it would be ideal to treat the other instruments as interruptible like the GHRS, the nature of the other instrument operations, combined with the magnitude of the required commanding

FIGURE 2. Variation of target visibility per orbit with declination and annual percentage. For example, a target at $\delta = 50°$ has orbits with visibility interval durations of 62 minutes or longer for 10% of the year. The range of variation is small for targets at low declinations.



changes, have made this infeasible. The next best alternative, which we have adopted, is to have Transformation construct SUs out of fixed-size components which are more *efficient* to schedule, but not more *difficult* to schedule *if planned in advance*.

There are three related changes to Transformation: target visibility modeling, exposure splitting, and exposure time adjustment. These changes are incorporated in Transformation 30.0, installed operationally on 23 April 1993, and in use routinely with these capabilities active since 20 May 1993.

2.1.1 Model the available time on target

Transformation has been upgraded with a simple model of target visibility time based on target declination (See Figure 2). For example, at $\delta = 50°$, 10% of all orbits during the course of a year have visibility interval durations of 62 minutes or longer, while 20% have durations of 57 minutes or longer, etc. Thus if Transformation is run with a "10% visibility percentage", SUs for targets at $\delta = 50°$ will be created on the assumption of a 62 minute visibility interval.

This change has critical implications for long-range planning: an SU Transformed with a "10% visibility percentage" must be scheduled in the ~10% of the year when the visibility interval is long enough for it to fit well. If such an SU is attempted to be scheduled in a week with less than 62 minutes of visibility time, then it either may not fit at all, or may fit very inefficiently. The situation is, however, not as strict as might be inferred from this one instance: for example, targets in the half of the sky at declinations $|\delta| < 30°$ have only about a maximum four minute variation in visibility time over the course of a year. If there are end-of-orbit gaps of this size in an SU, then the SU is effectively schedulable in a much larger proportion of the year than 10%.

2.1.2 Shape SUs by splitting exposures

Transformation has been augmented to "intelligently" split exposures or alignments where possible to maximize the contents of each orbit, based on the model of available target visibility time described above. This has the effect of eliminating or reducing gaps at the ends of each orbit, as well as often reducing the *total* number of orbits required to execute a given multi-orbit exposure sequence.

WFPC (Wide-Field / Planetary Camera) exposures (which are CCD camera readouts) are not subject to additional splitting out of concern for the impact of additional readout noise. Instead, the position of the cosmic ray split time is adjusted to allow each part of an exposure to fit into an orbit when it otherwise might not do so.

2.1.3 Shape SUs by shrinking or expanding exposures

Transformation has also been augmented to adjust exposure times by up ±20% (or by a specified amount) for certain types of exposures, in order to place more science into an orbit. It is important to realize that the Transformation algorithm operates in a way which makes *no adjustment at all* unless there is an increase in the exposure time efficiency of an orbit. No adjustments are made to the final orbits of single- or multi-orbit SUs. These features prevent the system from gratuitously shortening or lengthening exposure times to give the illusion of improving efficiency.

2.2 Spike: Commit SUs to weeks when they are efficiently executed

There are two major new capabilities added to Spike and installed with Spike 14.0, 10 May 1993. These are the "orbit packer" and the treatment of SAA-free orbits as a resource.

2.2.1 Analyze efficiency over time

The "orbit packer" is a component of Spike which estimates the duration of each SU if scheduled with the mean target visibility each day during the planning interval. It works by "laying down" the activities in the SU (GS acquisitions, alignments, SAMs, etc.) and seeing how long they take. The orbit packer considers the effect of target visibility time, shadow time, and alignment and SU duration and separation constraints. The effect of SAA impact is statistically modeled. The result is a measure of SU efficiency, i.e. the ratio of actual to minimum number of orbits, which can be used in Spike to control when SUs are committed in the long-range plan. In addition, times when SUs do not schedule at all are identified much more accurately than in the past, and these times can therefore be excluded from consideration during long-range planning.

2.2.2 Prevent the overloading of SAA-free orbits

The statistical model of the SAA used by the orbit packer makes it possible to identify SUs which can only be scheduled efficiently during SAA-free orbits in a particular week. If too many of these SUs are scheduled in a week (more than about half the week in duration), then SPSS will have no choice but to schedule some of them in SAA-impacted time: this will be inefficient at best, and may lead to significant gaps in the schedule. Spike now incorporates a limit on the number of SAA-free orbits consumed by the SUs committed to each week. Note that SUs which can be scheduled by hiding the SAA in occultation consume zero SAA-free orbits by definition; the number of these orbits committable in a week is therefore not constrained.

This approach to hiding the SAA is similar to that discussed in the analysis by Kinzel[6], except that it is implemented in Spike in terms of SU scheduling times instead of in terms of SU Right Ascensions.

2.3 SPSS: Sequence SUs for efficiency

There have been two distinct threads in the investigation of SPSS scheduling efficiency: one has been based on the existing SPSS AUTO automatic scheduling software, the other is a new tool designated CALOPT for “Calendar Optimization”. Fixes and improvements to SPSS / AUTO have been completed and were installed as SOGS 31.1C, 19 May 1993. The SPSS portion of the CALOPT software will be installed in July with SOGS 32.0, while the off-line portion remains available as an analysis tool.

2.3.1 Improve the SPSS automatic scheduler

SPSS incorporates an automatic scheduling mode which can be used to place SUs in the calendar. The automatic mode works following a “greedy” incremental scheduling strategy which makes a locally best decision based on a scoring algorithm. The scoring algorithm considers scientific priority, difficulty to schedule, total “non-science” (wasted) time, and several other elements. It was found that by fixing some problems with the existing software, and by changing the “non-science” component of the scoring to consider the *ratio* of useful to wasted time, the efficiency of the resulting calendars was significantly increased.

2.3.2 Exploit “global”information about SU schedulability

CALOPT follows a different approach and consists of two components. The first component runs in the SPSS environment: it steps through a calendar at short (e.g. 5 minute) intervals, attempting to schedule each SU at each time point t_i. The results may be that the SU does not schedule, or that it schedules over some time interval $(t_i, t_i + d_i)$. The results are written to data files, forming conceptually a large matrix where the rows are SUs, the columns are times, and the entries represent the duration d_i of the SU if scheduled at that time.

The second component of CALOPT is an off-line process based on the Spike scheduling software. The SPSS / CALOPT data is converted into a form usable by Spike. This hybrid system uses the SPSS / CALOPT SU times and durations, but dynamically models slews, SI reconfigurations, and FHST update time delays. It has been calibrated against SPSS*, but because it contains only simplified models, it is not expected to always find SU sequences that schedule directly in SPSS. The virtue of this hybrid approach is speed and flexibility, to support “what-if” analysis and the investigation of other scheduling heuristics, e.g. other greedy strategies than used by SPSS, or lookahead, backtracking search, and repair-based scheduling strategies.

At the present time the off-line Spike / CALOPT software is available only in a testbed (prototype) form.

3.0 Results

The improvements in the various software systems described in Section 2.0 have been exercised on a large number of proposals in order to evaluate their effectiveness and to ensure that there are no unexpected interactions. This section briefly describes the test program and then summarizes the results.

*This calibration process uncovered some pre-launch constraints that are no longer relevant which can be relaxed to improve scheduling flexibility.

3.1 The Efficiency Test Program

The test program was designed to simulate the main operations flows with a very limited investment of effort. All "Cycle 3"* proposals as submitted were included, excepting only parallel science, snapshot, and most moving target proposals (for which the effort to do the associated ephemeris determinations was not available). The test pool consisted of 144 General Observer (GO) and Guaranteed Time Observer (GTO) proposals, in the state just after Phase II submission (29 Jan 1993 for GTOs, 15 Feb 1993 for GOs), and before their consideration by the Proposal Implementation Team (PIT).

The major stages of the test were:

1. Transform all proposals with and without the efficiency improvement capabilities in order to identify problems and to assess the magnitude of possible efficiency gains.
2. Use Spike to schedule all SUs in an abbreviated (12-week) long-range plan.

*Cycle 3 refers to the observing period from about June 1993 through the HST Servicing Mission in December 1993.

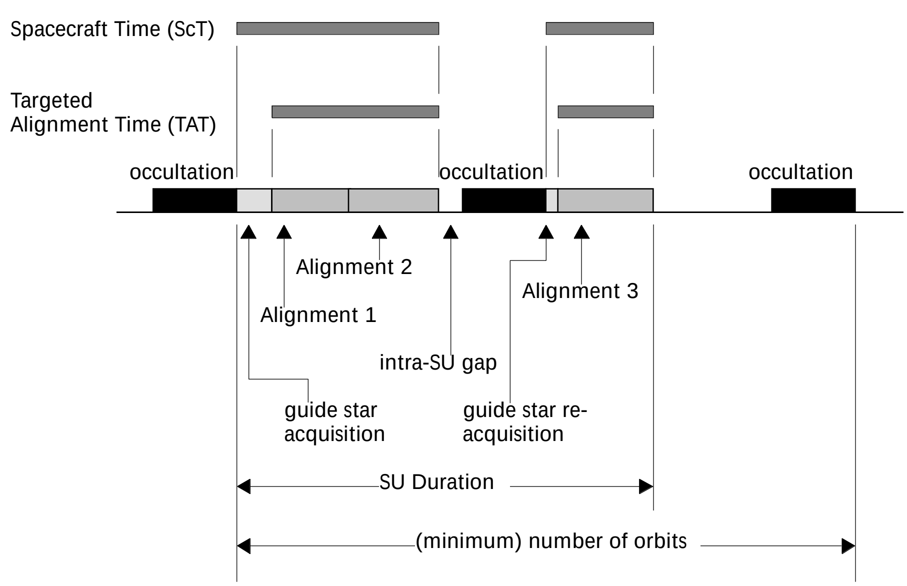
3. Use SPSS to schedule the contents of each Spike-committed week to ensure that each SU is schedulable when Spike specifies.
4. Run the automatic scheduling tools (SPSS AUTO, CALOPT) to investigate the scheduling algorithms and estimate the potential scheduling efficiency.

3.2 Transformation

There were several complete Transformation runs made with various parameter settings. The baseline run, referred to as ***BASE***, was with *none* of the improvements in splitting or exposure time adjustment enabled. The most aggressive run is designated ***EFF10***, to indicate that 10% visibility intervals were used (see Section 2.1). In the EFF10 run the exposure time adjustment was limited to ±20%, and all improved exposure and alignment splitting capabilities were enabled.

An important and useful measure of the minimum elapsed time required to execute a set of proposals is the "minimum number of orbits" (or simply "number of orbits"; see Figure 3), calculated for each SU by

FIGURE 3. Illustration of the terminology used in describing Transformation output and Scheduling Unit efficiency. A hypothetical two-orbit SU is shown, consisting of three "alignments" during which the exposures are taken. The instrument overheads are included in the alignment durations. See the Glossary in Appendix A for definitions.



estimating the time from the start of guide star acquisition to the end of the last activity (including an estimate of gaps due to target occultation), then dividing the result by the orbital period and rounding up to the nearest integer. This statistic has been calculated for the BASE and EFF10 runs and is shown in Table 1. By this measure the Transformation improvements lead to a 6% reduction in the minimum number of orbits required by the test proposal pool. As expected, the benefit is greatest for multi-orbit SUs: 41% of SUs greater than 3 orbits in size showed a reduction in the minimum number of orbits required. On the other hand, 53% of the SUs in the test pool required only one orbit and therefore were not adjusted at all. Note that the minimum number of orbits as computed by Transformation is not a good predictor of the overall scheduling efficiency of the proposal pool, since, e.g., the minimum may not be achievable for various reasons, the possible leftover time at the end of each SU may be usable for other programs, and continuous viewing zone (CVZ) opportunities are not considered. Two other relevant efficiency measurements from Transformation are the alignment and exposure time efficiencies, i.e. the ratio of these quantities to SU duration. These provide more direct insight into how the Transformation software is structuring Scheduling Units. The results in Table 1 show that the EFF10 run achieved ~10% improvements in both quantities. That these two should change together is not entirely obvious, since an *increase* in alignment splitting introduces additional overhead which tends to *decrease* exposure time efficiency. In the EFF10 run a total of 85 new alignments were created, a 3% increase over BASE. Nevertheless, the reduction in minimum required number of orbits more than compensates for the additional alignment overheads, and so the exposure time efficiency is significantly higher.

A separate measurement was made to determine how much of the savings from EFF10 was due to alignment and exposure splitting, and how much from exposure shrinking. The result is that splitting alone provides a 4.5% reduction in minimum number of orbits, compared to a reduction of 6% for splitting plus shrinking. The large majority of all SUs (85%) were untouched by shrinking or expanding, and only 3% were affected by as much as a 5% reduction in exposure time.

TABLE 1. Results of the Transformation runs on the Cycle 3 test pool: measures of efficiency improvement based on Transformation output products only.

Measure	Transformation Run: BASE	EFF10	Improvement
Total minimum number of orbits	1720	1618	6%
Targeted Alignment Time efficiency: (TAT / SU Duration)	50.0%	55.0%	10%
Exposure time efficiency: (Exposure time / SU Duration)	29.3%	32.1%	9.6%

3.3 Spike

All of the proposals processed in the Transformation BASE and EFF10 runs were given to Spike for long-range planning. A three-month period in 1992 was chosen as the planning interval and was broken into twelve one-week segments. A Spike automatic scheduling algorithm was run which tried to schedule each SU at its most efficient time, while taking into account time linkages and other constraints. The algorithm also tended to "front load" the schedule so that early weeks were more likely to be full. The commitments for BASE and EFF10 were done independently, and the SU lists were treated as "order forms" for the SPSS testing described below.

Not all of the SUs in the 144 proposals could be used in the Spike and SPSS testing. Of the original 791 SUs, 350 were found to be unsuitable for the test. These SUs were individually investigated and categorized; they included:

- pure parallels* (these SUs are attached *post facto* to the primary science schedule)

*A parallel observation is one which can be executed simultaneously with a primary observation. Proposals containing ***only*** pure parallels were excluded from the test; these additional parallels came from proposals which included both primary and parallel observing.

- SUs with long links (e.g. from precedence requirements) which prevented them from scheduling within the 12-week planning interval
- SUs with moving targets, or with special roll or window constraints which restricted the scheduling time to be outside the long-range plan period, or which required manual intervention to schedule
- SUs which were in violation of the solar exclusion constraint for the entire 12-week planning interval.

The SUs dropped from the remainder of the test program were not different in character from those left in: they were dropped only because the resources were unavailable to repeat the test in a way which would include them. There remained a total of 441 suitable SUs for SPSS testing.

3.4 SPSS

There were two parts to the SPSS testing: a schedulability test, and a "bulk" scheduling test. These are summarized in the following sections.

3.4.1 SPSS SchedulabilityTesting

The purpose of this test was to determine whether SUs allocated by Spike to weeks in the long-range plan were in fact schedulable by SPSS in the weeks specified. In case an SU was not schedulable, an intensive investigation was undertaken to determine precisely why not. The concern was that more efficient SUs would be harder to schedule and would therefore lead to an increased scheduling failure rate in SPSS.

The results of the SPSS schedulability testing are summarized in Table 2. The case of greatest concern is "false positives" from Spike, i.e. SUs that fail to schedule in their committed week. This ratio changed from 3.4% for the baseline to 5.2% for the EFF10 run. Each of the 23 EFF10 failures to schedule was investigated in detail: a total of 9 different causes were identified, distributed over Transformation, Spike, and SPSS. The single largest problem (40% of all cases) was a parameter error in Spike for the size of the terminator angle. Like several of the other problems, the effect was magnified due to the construction by Transformation of SUs that more tightly fit into the target visibility interval. The Spike problem was fixed in Spike 14.0, and at this point it appears that there is a negligible difference in schedulability between BASE and EFF10. The total number of "false positives" will be reduced by the detection and elimination of the classes of problems which have been revealed by this very exhaustive test.

TABLE 2. Results of the SPSS schedulability tests on the Cycle 3 test pool. Percentages are of the total of 441 SUs determined to be suitable for the test.

Measure	Transformation Run: BASE	Transformation Run: EFF10
SUs schedulable in Spike-specified week	367 (83%)	356 (81%)
SUs not committed by Spike, or linked to SUs not committed or scheduled	59 (13%)	62 (14%)
"false positives": SUs committed by Spike but not schedulable in the specified week	15 (3.4%)	23 (5.2%)*

*Reduced to 14 (3.2%) after resolution of the terminator angle problem.

3.4.2 SPSS bulk scheduling testing

The second set of tests focused on the total scheduling efficiency obtainable from SUs which were Transformed with exposure splitting and adjusting enabled, then scheduled by Spike with the orbit packer and SAA resource limit. Both SPSS AUTO and CALOPT were used in these investigations. Because the twelve weeks of the test long-range planning interval did not show significant oversubscription, four one-week oversubscribed calendars were constructed by combining SU data from a central week with the SUs from the two adjacent weeks. While this introduces some overlap in the scheduling candidates from one week to another, it permits an investigation of the effect of scheduling under high and variable levels of oversubscription (up to a maximum of about a factor of two).

SPSS currently measures efficiency in terms of "main fixed" time, i.e. the time in science activities that were scheduled as “mains” (as opposed to interleavers or parallels). This time can include internal calibrations, although most such exposures of this type are now done as interleavers or unattached parallels. For the bulk testing, internal and earth calibrations were excluded (except for the very small amount of time in internal calibrations embedded in GO and GTO SUs). So the resulting main fixed efficiency is quite close (within 1%) to the Targeted Alignment Time (TAT) efficiency (see Figure 3), the ratio of alignment time while the telescope is pointed at the sky to total elapsed time.

Another relevant efficiency measure is “spacecraft time” (ScT) efficiency (also shown in Figure 3), used by the HST Time Allocation Committee (TAC) for allocating observing time to proposers. The time counted by this measure includes Targeted Alignment Time, as well as guide star acquisition, small angle maneuvers, and other overheads specific to a proposal. These additional overheads are usually small, so that the main difference between TAT and ScT is guide star acquisition and re-acquisition times. A good empirical estimate of the conversion is:
ScT = 1.2 TAT.

Note that *exposure time efficiency* is not an important metric for SPSS scheduling. This is because the overheads for exposures are calculated and built into alignments by Transformation. They vary from one SI to another depending on exactly how each SI is operated. These exposure overheads cannot be changed in SPSS (except by re-Transforming). If exposure time efficiency was used to bias SPSS scheduling, the effect would be to prefer the Faint Object Camera (~40% exposure time efficiency) over the Faint Object Spectrograph (~35%) or the Fine Guidance Sensors (~20%).

It was determined early in the SPSS AUTO testing that problems were preventing the AUTO algorithm from operating to generate efficient schedules. The most significant problem was that observing efficiency was not properly factored into the decision of what SU to commit to what time. Once the greedy scoring algorithm was modified and AUTO was allowed to search through a substantial list of possibilities, the efficiencies went up significantly. The results are shown in Table 3, along with comparison results from CALOPT based on more exhaustive search using the global schedulability data as described in Section 2.3.

TABLE 3. Results of the SPSS bulk schedulability tests on the Cycle 3 test pool.

Test	Week	Targeted Alignment Time (TAT) efficiency BASE	EFF10
SPSS AUTO (best runs, nstep=50)	1	41.4	41.1
	2	45.2	47.6
	3	42.0	43.5
	4	41.3	42.3
	Mean	**42.5**	**43.6**
CALOPT (best runs)	1	40.4	42.2
	2	47.6	49.2
	3	44.8	45.4
	4	46.4	45.6
	Mean	**44.8**	**45.6**

The interesting results are:

- Typical TAT efficiencies are found to be ~44% with SPSS / AUTO and ~46% with CALOPT. Use of higher order search heuristics with CALOPT (not indicated in Table 3) has found only an additional ~1% of further gain (although one calendar did yield a TAT efficiency of just over 50%). This suggests that *the SPSS AUTO results are acceptably close to the maximum achievable.*

  For comparison purposes, the typical efficiencies of recent flight calendars (without snapshots, internals, or parallels) is about 30±5%, which is itself a significant improvement over the typical 25% efficiencies before mid-1992.

  For further comparison, a TAT efficiency of ~45% corresponds to a ScT efficiency of ~54%. This is significantly greater (by a factor of 1.5) than the 35% assumed for Cycle 3 observing. Note that it applies *before* snapshots, although there are few gaps which could be used for snapshot SUs which are as large as even 20 minutes.
- Run times for the most efficient SPSS AUTO runs ranged between just over 2 hours up to about 18.5

- hours, with the lower end of the range being more typical. These times are manageable on the current SPSS workstations, but it is clear that the new generation of DEC Alpha (AXP) workstations will significantly improve the ability of SPSS Operations to explore improved efficiency by routine use of AUTO. The AXP processors will provide a speedup by factors of ~5.
- There is a small but systematic difference between the BASE and EFF10 runs which indicates that the EFF10 runs are slightly more efficiently scheduled. The difference is only about one percentage point in these calendars, however it would be expected that the difference would tend to be masked by oversubscription (i.e. there is more likely to be an efficient SU available to schedule, even in the BASE run, since Spike is using the orbit packer to schedule efficient SUs in each week). It is also worth noting that the candidates allocated to each week were not required to be the same between the BASE and EFF10 runs, which introduces an additional source of uncertainty in the comparison.
- In the SPSS AUTO runs the amount of unocculted time impacted by the SAA was measured and found to be no greater than 3% (in the least subscribed week), and for three of the four weeks was less than 1%. This demonstrates that the systems are capable of "hiding" the SAA in occultation when provided with the opportunity to do so.

How much further can the efficiency be increased in SPSS? There are two major approaches: (a) increase and then use the amount of useful time on the sky, and (b) further reduce the number of gaps. Increasing the time on the sky is primarily a function of the mix of SUs in a week as determined by Spike. With the Spike orbit packer in place, continuous viewing zone (CVZ) and other efficient viewing opportunities will be strongly favored. Further reduction in gaps between or within SUs could be attempted, but a detailed examination of the gaps in the SPSS AUTO calendars suggests that there is not much to gain. For example, in the Week 3 EFF10 calendar in Table 3 (45.5% efficiency), filling in *all* the gaps larger than 15 minutes would add only *three points* to the efficiency, about the same as the increase from CALOPT. Trying to fill the smaller (< 15 minute) gaps — 60% of which are less than 5 minutes in duration — by further upstream Transformation changes would likely lead to diminishing returns by introducing more frequent scheduling failures. The conclusion is that there is not likely to be a further gain of more than about five points in the TAT efficiency: this corresponds to an additional ~10% improvement over the results in Table 3.

4.0 Conclusions

The various HST scheduling software systems as modified over the past year and currently installed are jointly capable of scheduling primary science with spacecraft time (ScT) efficiencies in excess of 50%. This level of efficiency represents an increase by a factor of ~1.5 over the currently realized primary science throughput of 30-35%. The 50% efficiency level is an indicator of what the software systems are capable of achieving under nearly ideal conditions and when observing efficiency is the primary goal. Operational realities will limit to some degree our ability to actually achieve efficiencies of this magnitude. The disruptive effects of spacecraft problems and consequent replans, or of software problems and workarounds, will tend to act to reduce the actual observing efficiency. Furthermore, the competition between efficiency and other valid and important goals (e.g. scientific urgency or priority, meeting processing deadlines) will also tend to reduce efficiency. Consequently the results reported here are likely to be a practical upper bound on the achievable efficiency. This bound provides important insight into what kinds of efficiency goals could be adopted for HST observing. Further technical improvements are not likely to increase this bound by more than a few percentage points.

References

1. Miller, G. and Johnston, M. 1991. "A Case Study of Hubble Space Telescope Proposal Processing, Planning and Long-Range Scheduling." Proc. Computing in Aerospace 8, Baltimore, MD (American Institute of Aeronautics and Astronautics), 1-13.

2. Jackson, R., Johnston, M., Miller, G., Lindenmayer, K., Monger, P., Vick, S., Lerner, R. and Richon, J. 1988. "The Proposal Entry Processor: Telescience Applications for Hubble Space Telescope Operations." Telematics and Informatics 5: 107–124.

3. Rosenthal, D., Monger, P., Miller, G. and Johnston, M. 1986. "An Expert System for the Ground Support of Hubble Space Telescope." Proc. 1986 Goddard Conf. on Space Applications of Artificial Intelligence, Goddard Space Flight Center, Greenbelt, Maryland

4. Gerb, A. 1991. "Transformation Reborn: A New Generation Expert System for Planning HST Operations." Proc. Proceedings of the 1991 Goddard Conference on Space Applications of Artificial Intelligence, Goddard Space Flight Center, Greenbelt, MD 45-58 (reprinted in Telematics and Informatics 8: 283-295.)

5. Johnston, M. and Miller, G. 1993. "Spike: Intelligent Scheduling of Hubble Space Telescope Observations." in Intelligent Scheduling, ed. M. Fox and M. Zweben (Morgan Kaufmann: San Mateo), (in press).

6. "Observations and Suggestions for Observing Efficiency and the SAA" W. Kinzel, 17 August 1992, STScI memo.

App. A. Glossary and Acronyms

Term	Definition/Comments
Alignment	A component of a Scheduling Unit which usually consists of a single telescope pointing. An alignment can contain one or more exposures. An SU is an ordered sequence of alignments. Alignments are defined by Transformation.
AUTO	The SPSS automatic scheduling algorithm, based on a "greedy" strategy.
CALOPT	Calendar Optimization: an approach to sequencing SUs in SPSS which makes use of global rather than local schedulability information.
FHST	Fixed Head Star Tracker, used for attitude updates before GS acquisitions. May introduce gaps between adjacent SUs.
GHRS	Goddard High-Resolution Spectrograph: the only one of the scientific instruments on HST which can be scheduled as "interruptible"
GS	Guide Star
Main fixed	An accounting measure of time or efficiency which includes alignment time for SUs scheduled as SPSS "mains". Excludes GS acquisition times as well as occultations and deadtime. Can include internals, earth calibrations, etc. if they are scheduled as "mains".
SAM	Small Angle Maneuver
ScT	Spacecraft time: an accounting measure of time or efficiency which includes alignment time, GS acquisition, and other required intervals (e.g. small angle maneuvers) but which excludes occultations and deadtime (e.g. SAA). See Figure 3.
SMS	Science Mission Specification: the detailed spacecraft and instrument command request schedule which is produced by SPSS as the ultimate output product of short-term scheduling.
Spike	The HST long-range planning system.
SPSS	Science Planning and Scheduling System: the HST short-term scheduling system.
SU	Scheduling Unit: a collection of exposures scheduled as one "entity" in SPSS and Spike. SUs are defined by Transformation or by a manual override of Transformation's automatic rules.
SU Duration	An accounting measure used for determining how much time Spike should send to SPSS in a week. Calculated by Transformation as the minimum estimated time from start of first GS acquisition to end of last alignment in the SU. See Figure 3.
TAT	Targeted Alignment Time: an accounting measure of time or efficiency which includes time on external targets only. Does not include GS acquisitions, SAMs, or any dead time. See Figure 3.
Trans	Transformation: the software system which converts HST observing proposals into the data structures which allow them to be scheduled by Spike and SPSS.
WFPC	Wide-Field / Planetary Camera: the CCD camera instrument on HST